

BAB IV PEMBAHASAN

4.1. Kesimpulan

Berdasarkan rumusan masalah yang telah dibahas, berikut adalah kesimpulan yang dapat diambil dari penelitian ini:

1. **Penerapan Data Driven Programming (DDP) dalam pengelolaan data kartu, area, dan dialog NPC pada game Ghosty Ghost:**

Penerapan DDP berhasil dilakukan melalui penggunaan *Data Asset* di *Unreal Engine*. *Data Asset* memungkinkan penyimpanan dan pengelolaan data secara terstruktur dan modular, sehingga mempermudah proses pengolahan dan pembaruan data tanpa harus mengubah kode program secara langsung. Hal ini memberikan fleksibilitas dalam mengatur atribut kartu, pengaturan area, serta dialog NPC, yang dapat diubah atau ditambahkan secara dinamis oleh tim pengembang maupun desainer tanpa keterlibatan langsung dari programmer.

2. **Alur data dari tahap perancangan hingga digunakan dalam gameplay dengan menggunakan *Data Asset* di *Unreal Engine*:**

Alur data dimulai dari tahap perancangan atribut dan struktur data yang sesuai untuk kebutuhan gameplay. Data kemudian diimplementasikan ke dalam *Data Asset* sebagai entitas yang dapat digunakan ulang (reusable) di berbagai komponen game. Dalam *gameplay*, *Data Asset* diakses secara langsung oleh sistem permainan untuk memuat informasi kartu, area, dan dialog NPC. Proses ini memastikan bahwa data tetap konsisten dan terpusat, sekaligus mempermudah debugging serta integrasi data baru ke dalam game.

4.2. Saran

Berdasarkan hasil penelitian dan penerapan *Data Driven Programming* (DDP) dalam pengelolaan data pada game Ghosty Ghost, terdapat beberapa saran yang dapat dijadikan acuan untuk pengembangan lebih lanjut:

1. Peningkatan Fleksibilitas Pengelolaan Data

Usulan untuk mengintegrasikan tools atau metode baru yang dapat mendukung pengelolaan data lebih fleksibel, misalnya memanfaatkan *API* eksternal untuk sinkronisasi data.

2. Pengembangan Fitur Baru

Saran untuk menambahkan fitur-fitur baru pada game Ghosty Ghost, seperti Variasi area permainan dengan lebih banyak elemen dinamis dan Dialog *NPC* yang interaktif berdasarkan keputusan pemain.

3. Optimasi Performa Sistem

Rekomendasi untuk mengoptimalkan performa sistem, seperti pemuatan data secara bertahap (*lazy loading*) untuk mengurangi beban awal memori dan penyesuaian algoritma pemrosesan data agar lebih efisien.